

VIII: A Brief History of RoboWar

Extracted from the MacOS version of RoboWar 4.5.2 Instructions, copyright David Harris and Lucas Dixon

Many moons ago, the world was lacking RoboWar. Then one crisp spring night a bunch of fanatical teen-age computer programmers were sitting around at a meeting of the Ridgecrest IEEE Student club tossing out programming ideas and dodging the shrapnel. Sick of hand-eye arcade games, the concept of robots fighting without user intervention grabbed their attention. Thus was born RoboWar.

The computer club members debated many ideas. At first they considered placing robots on a board full of obstacles and challenges. They thought about building robots by dragging icons about to form various subsystems. Jon Richards was the first to prototype RoboWar code on a foul MS-DOS clone. Later that summer at Caltech, David Harris saw an early version of another robot-battle style game being developed by students there. He liked the concept of the programming language for robots and saw a number of other ideas for improvement.

In the fall, while climbing Dragon Peak out of Onion Valley in the Sierra Nevada, David was hit across his head with a fascinating, efficient, and easy to implement programming language based on Reverse-Polish notation. Fortunately, another computer club veteran, Ralph Giles, was also on the hike and the two worked out the details of the language, with inspiration from the HP-28 calculator language, C, Pascal, and assembly language in addition to their own foul concoctions. Fortunately, they wrote a complete specification of the interpreter, for, although Ralph spent a bit of time working on it on another clone, it wasn't until December that David began to implement it on a real machine.

David first designed an interpreter and compiler that ran, albeit clumsily, on a Silicon Graphics workstation. Then he ported it to the Macintosh, built up a user

interface, and spend most of Christmas vacation chasing bugs. When the program worked reliably, progress greatly slowed, for he spent extensive time 'playtesting' instead of coding.

Since that point, robots have evolved through a number of stages. This history details the development of code by the IEEE club members. Doug Harris, David's demented brother, also built a number of robots, ranging from unreliable to excellent in quality, that followed some convergent and some divergent branches of evolution. The IEEE club robots have come through six generations:

First generation robots include MoveBot (listed below) and DumBot. They were written to test the interpreter on the Silicon Graphics. They duel nicely but fairly randomly. Each was generally less than a page in length and wasn't smart enough to kill its opponent in a single shot. TimBot (also listed below), a simple but effective robot, led the second generation of robots into existence. Written by Tim Seufert on the Macintosh, it locked onto stationary robots and blew them to tiny bits. Originally, it used less energy in its shots and took longer to eliminate its opponents, but still, TimBot made completely stationary robots unviable.

In response to TimBot, drooling hackers started to produce a third generation of robots. These typically moved about the board. One interesting pair of third-generation robots were Coroner 1 and 2. These 'bots moved to opposite corners before shooting at any targets they saw. If they began taking damage through their shields, they would flee to the opposite corners.

Matt Sakai changed the entire course of robot evolution when he brought Silo IV to a meeting on a nondescript floppy disk. Silo IV, the one and only fourth generation robot, moved about the board and almost unfailingly destroyed its first, second, and third generation opponents. For a time, the IEEE group was stunned; David even added some more features to the interpreter that robot designers might try because Silo IV seemed to be the ultimate robot in the evolutionary tree.

Nonetheless, after a few weeks, numerous other robots using similar tactics sprung from the silicon mind. These fifth generation robots, the so-called 'Silo clones', eventually managed to equal, then best Matt's Silo IV. Among the Silo clones were Robot B2 (the product of David Wasserman's nightmares), TimBot IV

(at first the joke of the IEEE group, later a reasonably effective robot), Freud (most famous for his remarkable icon), and Blade (the best Silo clone yet developed). To test these robots, the IEEE members ran extensive combats and developed the 'TimBot Test:' how many plain-vanilla TimBots on one team can one robot defeat at one time in at least 60% of the combats? Most Silo clones could almost always defeat a single TimBot and could defeat two more than 60% of the time.

Fortunately, before things became too dull, robots using other strategies began to appear. Among these sixth generation robots were Lewis Girod, Aeneas II, and Pearl. Aeneas broke the three TimBot barrier. Unfortunately for his creator, Pearl appeared soon after, usually capable of overcoming five other TimBots teamed together in a battle! Pearl, another product of Matt Sakai, is the current champion.

What is the future of RoboWar? Pearl appears to be very good but Silo at one point also seemed unbeatable. Few robots have been written to work effectively together on teams; few use the communications or advanced projectile detection capabilities. Perhaps somebody will develop tracking algorithms or discover another excellent algorithm lurking just out of sight. The group has proposed a number of questions about robot behavior that they still have not answered, so there is still ground to explore. For the novice, RoboWar has much excitement waiting as one learns to program and to overcome each generation. For the expert, too, the horizons of RoboWar are still beckoning.

In such a state RoboWar rests April 16, 1990. I will be sponsoring a RoboWar tournament in early June in which the best robots of each contestant may compete for glory and riches. See the about box for more details. Also, please pay your shareware fees if you enjoy RoboWar. I put a great deal of time and effort into programming. Since the \$10 fee is so much less than the cost of a professionally marketed game, I would appreciate the money.

History Continued. On June 1st, (tonight, as I write) we are collecting the robots for the tournament. We received 11 entries at \$2 each. Matt introduced his reputedly fearsome seventh generation 'Religion' robots-Hinduism, Judaism, and (The Great God) Anything, replete with wild comments as well as powerful new designs. Yes, there are robots better than sixth generation Pearl clones! We are waiting for the robots from Northwestern University (hopefully Jon will send some)

and plan to hold the competition very soon. YAWP!

Continued: The RoboWar competition is complete! Twelve robots entered; their names, owners, and comments are listed below:

Robot	Creator	Comment
ShazBot	Dave Gangon	The Physicist
Line II	Wesley Voshell	De Best
TimBot V Mark II	Tim Seufert	Kills Strafers
Hinduism	Matt Sakai	Strafer that'll kill TimBot anyway
Judaism	Matt Sakai	No comment.
Anything	Matt Sakai	Convert or Die!!!
Silo Plus	Matt Sakai	Like the icon?
Blade	Ralph Giles	The old battle-ship
Sch'n	Ralph Giles	How much potential is realized?
Line	Matt Mann	??????????
Appletree	David Wasserman	Cannon fodder
Tower Bot	Doug Harris	Have a nice day

Jon never sent any robots (finals were the week before the robots were due) but Doug entered his Tower Bot which had a nasty habit of overflowing the stack just after the 3500 chronon battle limit was over and thus forcing the battle rosters to be redone. Fortunately, with the help of Ralph's Mac Ilci, a version of RoboWar using the 68881 instructions, and the automated combat features, the tournament only took a morning to complete.

First Place goes to Silo Plus, Matt's seventh-generation shapechanger. ShazBot took second place with a good algorithm to escape wall huggers and to track Silo

clones. Finally, the shield-bearing Silo clone TimBot V Mark II surprised us all and took third place.

From this contest, we find the old guns still do fairly well. Blade, the fifth generation robot, far outperformed his cousin Sch'n, the sixth-generation perimeter 'bot. Matt had a good design in his seventh-generation robots, though. They were typically very efficient wall-huggers that changed behavior (thus the name shapechanger) as the game progressed.

Over the next several months, RoboWar received only sporadic work. The 500 instruction limit was clearly a limiting factor for new robot development, so raising the limit to 5000 was a crucial part of the new RoboWar 2.0 program. Also, the Central Control metaphor, though intuitively simple, required a large number of mouse clicks to navigate and violated the standard Macintosh interface pretty badly. The biggest change to RoboWar 2.0 was a reorganization of the user interface to be (hopefully) much faster to use.

David also began receiving shareware checks and letters about RoboWar. Letters from new RoboWar addicts asking for new features were very exciting and spurred on development of RoboWar 2.0. The program was revised by the end of January but these instructions needed a rewrite. Unfortunately, with the semester starting, David didn't have time for the full rewrite that the instructions deserved, but hopefully this document is accurate and clear enough to suffice for the present.

What is the future of RoboWar? RoboWar 2.0 is so full of new features that I'm afraid some will turn out to be Intel Features (a.k.a. bugs). Over the next few months, I'm planning to collect bug reports and post fixes on the network (at sumex-aim.stanford.edu for all you Unix types out there). David had a number of new features (a debugger in particular) that he had wished to include in Version 2.0 but lacked time to complete. These new features will appear in Version 2.1, 2.2, etc. as time permits.

And now for the results of the Second RoboWar tournament. This tournament's entries were quite exciting, stretching RoboWar 1.5.1 to its very limits. Eighth generation robots who specialized in evading enemy shots were common in this

contest. The contest had sixteen individual entries plus David's old standards, Freud and Aeneas III, placed in the arena for comparative purposes. Rammer AMT, by Jon Newman, scored a clear first place victory. Chicken & Corn II, by Tom Morrison, overcame his moderate performance in the initial rounds to place second. Finally MX' III, by Doug Harris, placed third with his TacNuke superbombardment. Mx3 and Mortician', a team jointly written by Doug Harris and Tim Seufert, clearly won the team competition. Patriot, by Wesley Voshell, won the Icon Competition with a marvelously drawn picture of a ship. Congratulations to everyone who entered the contest. This competition featured some of the best robots I had ever seen. With the new flexibility of RoboWar 2.0, I expect to see even more exciting robot designs.

In the spring of 1991 I ran a class on RoboWar at MIT for junior high and high school students. Each week the students would program robots and bring them to class to compete with their fellows. I observed how the robots evolved over time; each week there would be minor improvements over the previous week's robots but every once in a while, somebody would introduce a significantly different robot technology (a "paradigm shift") that could defeat most all previous robots. The new robot would spur another series of incremental improvement until finally there was another paradigm shift. I also found that the chance for students to interact and observe each other's robots helped the students to develop much better robots. By the end of the class they had programmed robots that could compete well with the entries to the second tournament. RoboWar proved to be a valuable educational tool for teaching programming techniques to fairly young students. I would like to thank the members of my class for all the bugs they helped me track down and for the insight they gave me on RoboWar learning patterns.

In July, Dave 'Rasferet' Blumenthal, a Cornell student, entered the RoboWar Hall of Fame by cracking the RoboWar password system. As promised, Dave is now due for infinite glory and undying mention in the sacred RoboWar Instructions. Dave has also made numerous suggestions and bug reports that have helped my development of RoboWar very much.

In May 1992, the Fourth RoboWar Tournament was held, featuring some extremely well-designed robots. Doug Harris, whose clever designs in the past

were fatally wounded by bugs, won both the individual (Pacifist Penguin), and (with the help of a robot by Tim Seufert) team (Miracle Penguin & Terminator II) prizes. Robert Hogg again demonstrated his RoboWar mastery taking second place with Beholder Jr., tying for third with Biosphere, and winning the icon contest with the utterly cute Lewis. Eric Foley's Orb of Doom also tied for third in the individual contest. This time there were a number of effective teams; moreover, many of the robots had extremely good icons and not a single entry performed very badly. Jesse Barnum discovered a remarkable new way to cheat, but the bug was fixed before the tournament so he just receives an honorable mention.

Two weeks after the tournament, misplaced mail surfaced at the East Campus desk. Enclosed was Jeff Rommereide's Excelsior, his Fourth RoboWar tournament entry. As the results had already been mailed out, I was unable to enter Excelsior in the tournament, but I staged a mock tourney anyway. Excelsior claimed first place! This was quite tragic; Jeff had invested a great deal of effort in new technology for this robot and it performed exceptionally well. Thus, I promised Jeff a glowing description of Excelsior in this manual and expect that it will be entered in the next RoboWar tournament to challenge hackers. Congratulations, Jeff: may Excelsior claim eternal glory in the halls of RoboWar Heroes!

A few weeks later, the impossible happened: I finally got around to implementing the RoboTalk debugger! YAWP!

A large contingent of RoboWar hackers negotiated a site license at Carleton College and ran their own local tournament in October 1992. Tom Hayes sent me some of the results; Nereid was the winner and Leech, an annoying 98 instruction bot, took second. Little VT 002 won the Little League division. I do not have any of the robots or any more information, but you can enquire with Tom at THAYES%ADMIN1@carleton.edu. If any other groups are interested in RoboWar site licenses or run noteworthy local tournaments, drop me a letter...

The Fifth Tournament was somewhat smaller, but featured a number of excellent robots. The individual winners were (from first to third): Excelsior by Jeff Rommereide, Lug by Robert Hogg, and Orb of Doom II by Eric Foley. Robert also

won the team contest with \$*Lucky Flea*\$ and \$*Nuclear Flea*\$; these and other suicidal destroyer teams dominated the contest. In the interest of promoting strategy over luck, suicidal destroyers will be banned from future tournaments. Eric won the first titan competition with the dreaded Krulockh Lord and Doug Harris was winner of the little leagues with Wimp. Finally, Dirty Lewis tickled the funny bone of the judges and claimed supremacy in the icon contest!

After a long hiatus from RoboWar programming (taking a heavy load at MIT does wonders for gobbling up all free time), I finally got back to adding my favorite features from people's wish lists during finals week of Spring 1993. These features appear in RoboWar 2.4. They include the Recording Studio for adding sound to robots, interrupts (which may revolutionize robot design), and lots of fixes of small but annoying bugs reported over the last year. Thanks to everyone who contributed suggestions or bug reports; your ideas help make RoboWar a better program.

The Sixth Tournament, held June 1993, claimed the distinction of being the first International Tournament. Three of the twelve people entering the contest came from overseas and many of their robots did very well. The Darling series of profoundly aggressive robots ushers in a new generation of robot designs that has never been seen before. The individual winners were (from first to third): Sylvestre by Antoine Duchateau, Darling, by Stephen Linhart, and Dragon Knight by RoboMaster Eric Foley. Stephen Linhart also won the team and Little League categories with a pair of Darlings and with Lil Darling, respectively. RoboMaster Douglas Harris won the Titan competition with the latest of his Penguin series, z Death Penguin. Finally, Eric Foley won the icon contest with his all-seeing and unpronounceable Eye of Shorshirsh.

RoboWar never was released as version 2.4; instead, I took the leap up to version 3.0 by adding automatic math coprocessor support, interrupts, and so forth. Version 3.0 should be released in September 1993.

RoboWar 3.0 introduced almost as many new bugs as new features. Thanks go to the many RoboWar hackers who sent detailed bug reports that have helped me squash these problems. Some names particularly worthy of recognition are Stephen Linhart, David Pokorny, and Tim "the enchanter" Mattox. Keep those bug

reports coming!

Version 3.1, released at the end of January, 1994, fixes many of these bugs. Unfortunately, the dreaded Thesis prevented me from adding any of a long list of desired new features, but one of these days, Version 3.2 should come out with any remaining bug fixes and with a couple of minor new features.

At this point, the artistry of Brian and Greg Parker should be mentioned in the RoboWar Hall of Fame. These two intrepid young RoboWar hackers drew nifty scenes of robots blasting each other to oblivion on the envelopes in which they sent various pieces of correspondence. Nice drawing!

Andrew Lindsey's "His All Seeing Eye III" (entered in the 7th tournament) takes advantage of an undocumented feature and a glitch in RoboWar to pull a really neat trick. Try tracing its code in the debugger and watch what happens!

In May 1994, I finally upgraded from trusty old Mac II to a spiffy new PowerMac 8100. Using the excellent Metrowerks compiler, I ported RoboWar to run native on the PowerMac. Video isn't much faster, but when the battle isn't displayed, it runs up to 30 times faster than it did on my old machine! Nice job, Apple!

The Eighth International RoboWar Tournament, held in July 1994, was the largest and best so far. Interrupt bots have come of age and many clever strategies are employed.

RoboWar 4.0 evolved over the summer of 1994. The Metrowerks compiler is an excellent product and the programmers obviously are craftsmen who love their work. Unfortunately, it isn't as thoroughly tested as other compilers and has produced bad code on occasion. Metrowerks is very good about fixing bugs that are reported, but I am afraid that sneaky bugs might be left in RoboWar that I haven't caught. Thanks go to Tim Seufert and Ben Matasar for beta testing RoboWar 4.0 and catching lots of bugs!

One of the most interesting (or perhaps least interesting, if you have a slow Mac) results of the Eighth International Tournament was the evolution of cooperation among robots. The scoring system gives one point to any robot that survives in a duel. RoboWarriors have gradually realized that killing the opponent doesn't help

one's own score; hence a number of robots have emerged that don't attack unless they sense aggression from the other party. Eric Foley describes these robots as the "Alliance." Such alliance members can often outscore even the best aggressors simply by scoring lots of points in each round they encounter fellow alliance members. This situation is a variant of the Prisoner's Dilemma and also has many parallels in the real world, where peaceful nations can advance economically while aggressive ones mortgage their future for new bombs. It is fascinating to see such real world cooperation emerging from a simple simulation like RoboWar where no communication is possible save for observing the opponent's actions. Unfortunately, the Alliance leads to many long boring rounds where the robots simply watch each other for 1500 chronons before a draw is declared. To appeal to our lust for senseless virtual violence, the scoring system may change in the future to reward effective killers as well as robots that live to a ripe old age.

RoboWar 4.1 features new registers and operators and an improved tournament utility. Jeff Lewis of Bakersfield, CA deserves credit for pointing out lots of bugs in the old tournament routines and suggesting the nifty new tournament interface. The history registers evolved while contemplating extended games of Prisoners' Dilemma while hiking Mission Peak. They should open up lots of creative strategies, but risk increasing the complexity of a world-class robot. Thanks to Eric Foley's urgings and lots of boring Alliance members in the Ninth Tournament, a new aggressive scoring system has been instituted that rewards robots for kills as well as survival.

RoboWar 4.1.1 and 4.1.2 mostly fix bugs. In September, 1995, I moved to Stanford to begin my Ph.D. RoboWar business has picked up in 1995 (largely thanks to the growth of the net) and my mother Sally Harris has generously offered to handle registrations so I'll have more time to design microprocessors. The 4.1.2 release reflects the new RoboWar HindQuarters address (back at the RoboWar birthplace in Ridgecrest, CA) and a new email address that will hopefully be processed more rapidly than my Stanford one.

The Tenth International RoboWar Tournament continued to be dominated by Jean-Francis Lechat, the Napoleon of RoboWar. Jade held out in the lead and

Carne demonstrated a very novel use of drones. For fairness, drones and lasers will no longer be allowed in future tournaments (because Doug Harris complained they violate the "no undocumented features" rule). Who will devise a robot clever enough to overcome Jean-Francis' minions in the next tournaments?

The Tenth Tournament also had a loophole in the rules that allowed self-destructive robots in the Individual competition to rack up many points in group combat. To patch this loophole and encourage survival in general, RoboWar 4.1.2 only awards points to robots for kills if the aggressor is still alive at the time of the kill. Also, some robots used lasers and drones. These weapons are now in the category of "undocumented features" and are banned from future official tournaments. Finding them is a challenge for the hacker; please do not ask me about how to enable or use them or to bring them back into the game as an official feature.

RoboWar 4.1.2 was finally released in December 1995 after I tracked down and squashed an annoying bug that caused projectiles to move at different speeds depending on where they were in the arena. The bug was difficult to find because it only happened on certain Macintoshes and because some of the reports about the bug seemed to have incorrect information. Moreover, the bug did not occur on my development machine, so I had to find a machine where it did happen and port my development tools. In the end, the bug turned out happen on 68K Macs with floating point hardware; it was due to letting the floating point routines get out of date with respect to the normal routines. It should be fixed now. RoboWar 4.1.2 also fixes a systematic bias that caused robots early in the alphabet to do slightly better in tournaments.

Thanks to RoboMasters Eric Foley, Jean-Francis Lechat, and Greg Parker, as well as Lucas Dixon and numerous other RoboWar hackers, for detailed bug reports that greatly stabilized RoboWar 4.1.2.

Eric Foley has developed a very nice web page with a taxonomy of RoboWar strategies. His categories are much more up to date than this history. It can be reached from the RoboWar web page.

The Eleventh Tournament chronicled the rise of Dashers, robots which capitalize

on the aggressive scoring rule. In particular, the SpamBot series of dashers built by Colin Jaffe dominated the tournament. Many RoboWarriors became discouraged, fearing that the Twelfth Tournament would feature only incremental improvements in dasher technology and that new strategies were unlikely to arise. Denys Seguret surprised the RoboWar world with his innovative Dialectix robot which soundly crushed the dashers. Meanwhile, Eric Foley swept the titan round with his horde strategy of robots which recognize their friends. Hordes were novel once, but will be discouraged in future tournaments because they essentially amount to buying the victory with a large entry fee. The Thirteenth Tournament should be exciting as dashers and anti-dashers and new strategies vie for the title.

RoboWar 4.2 was released in January 1997, offering battle replay (for easier testing) and various minor improvements and fixes. Thanks to many who suggested improvements and pointed out bugs, including Colin Jaffe, Thong Nguyen, Lucas Dixon, Andrew Clinton, Paul Olson, Eric Foley, Steven Merrill, Amn Arah J. Leonard (who gave the instructions a much-needed proffreading), Bryant Brandon, Tom Livak, and Denys Seguret. At this point I am trying to stabilize RoboWar. Many people have sent in suggestions about interesting weapons and modifications to the arena, but I am not currently considering such changes because they will upset the balance currently existing in the arena.

After at least four years of promising color icons and being too lazy to deal with the finer points of Color QuickDraw, I took up Lucas Dixon's offer to implement them in spring 1997. He did a pretty slick job with it, leading to RoboWar 4.3. The difficulties of managing two versions of the code lead to some bugs (hopefully we've tested 4.3 sufficiently, but I never know until I turn the bugs loose on the world), so I'm not too eager to farm out development in the future, but Lucas Dixon and the beta testers deserve thanks for a much-needed job. Incidentally, Lucas added a default icon of a warrior for new robots. The icon is from Realmz, a fantasy game from Fantasoft. Thanks to Fantasoft for permission to use the icon; check out their web site at <http://www.fantasoft.com>. RoboWar 4.3 also fixes a bug involving negative energy which was exploited in an unofficial tournament. Remember that exploiting bugs in official tournaments is illegal and will result in disqualification (and the Golden BozoBot Award if you try to hide it).

Eric Foley hosted the Thirteenth International RoboWar tournament. There were fewer entries than usual, but the quality was very high. Eric swept a bunch of the top positions (no, he didn't cheat!) elevating himself to holding the all-time largest number of RoboWar awards. He's also developed a thorough history of RoboWar tournaments which can be reached from the RoboWar Web page.

Unfortunately, three more individuals receive the Golden BozoBot Award. After being clearly told that exploiting the negative energy bug of RoboWar 4.2.1 would be illegal in the Thirteenth Tournament, Nathan Paymer did so anyway and tried to conceal his action. Ever-vigilant Foley caught the cheating and nominated the robots for the BozoBot award. More seriously, Matt Wellstein's Twelfth Tournament entries (Eliminator, Shimmiebot, and Razorback) plagiarized code extensively from Cloak Folk's Revenge, Darling, and Vortex of previous tournaments. Matt's comments suggested that he had invented the code himself, but in actuality it was copied almost completely. I sent mail to Matt requesting that he apologize to the author, but got no reply. Matt receives one of the worst Golden BozoBot Awards in the history of the game for stealing code, claiming it as his own, and declining to apologize. Thanks to the alert RoboWarrior who pointed out the theft. Similarly, David Eisenstat stole the exact code from Jade and entered it in the Eleventh Tournament under the name Dave, forever disgracing the noble name.

RoboWar 4.4 is a transition, which fills me with mixed feelings. I've been maintaining and improving RoboWar for eight years now. It's been very exciting to meet so many RoboWarriors around the globe. It's also been profitable enough to pay for my development computers and software, though shareware is never a way to get rich. I've watched RoboWar evolve as I've moved from high school to MIT to professional engineering and back to graduate school. It's a perfect lesson in large-scale software engineering and in the challenges of maintaining code over eight years of changes. My biggest regret is hard-coding lots of calls to the Macintosh OS into the code, which made RoboWar run fairly fast on the Mac Plus but also means RoboWar is nearly impossible to port to other platforms. I've grown too lax with upgrading RoboWar, though, so for the sake of the game I've decided it is time to pass it on to new blood, namely Lucas Dixon. Lucas is a bright and enthusiastic hacker from Scotland who's agreed to continue improving

the game for some time to come. I'm sure you'll read his musings in future entries to this history.

As my first entry into the RoboWar history I'd like to thank David Harris for doing such a great job. His name will remain a legend in RoboWar for all time.

After a couple of minor fixes producing RoboWar 4.4.2, I decided to take on the Drafting Board in a complete rewrite to use the WASTE text engine. This with a few other changes and a preferences dialog box, and a Major, at the time, and much disputed, change to the Game Engine to disallow robots from being able to move and shoot in the same chronon made RoboWar 4.5. Which was more buggy than I had hoped for, and had some unexpected conflicts with versions 7.5.1 to 7.5.3 of the MacOS. Tournament 16 brought new life to RoboWar having stopped the game killing dasher paradigm.

Started in late 1998 a complete rewrite of RoboWar in C++ is making it possible to port RoboWar to Windows (urk!) and Linux (Yay!). This is expected to have it's first release in the summer of 1999. I'm also hoping that it will make RoboWar more stable, easier to change in the future and will also let new languages be developed. I'm particularly interested in Prolog, a language much-used by the University of Edinburgh's Artificial Intelligence department where I am studying.

Remember that RoboWar thrives because of you, the clever RoboWarriors who continually seek to overcome one another with smarter, stronger, and more efficient robots! Keep up the innovation that makes RoboWar so unique among computer games!